APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/764,205	01/23/2004	Atul Mukker	03-2586	8069

24319   7590   07/10/2008

LSI CORPORATION
1621 BARBER LANE
MS: D-106
MILPITAS, CA 95035

EXAMINER
SAEED, USMAAN

ART UNIT	PAPER NUMBER
2166	

MAIL DATE	DELIVERY MODE
07/10/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _21 April 2008_.
2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.
3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1-8 and 12-22_ is/are pending in the application.
   4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5)☐ Claim(s) _____ is/are allowed.
6)☒ Claim(s) _1-8 and 12-22_ is/are rejected.
7)☐ Claim(s) _____ is/are objected to.
8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.
10)☒ The drawing(s) filed on _23 January 2004_ is/are: a)☒ accepted or b)☐ objected to by the Examiner.
   Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
   Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a)☐ All   b)☐ Some * c)☐ None of:
      1.☐ Certified copies of the priority documents have been received.
      2.☐ Certified copies of the priority documents have been received in Application No. _____.
      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
   * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☐ Notice of References Cited (PTO-892)
2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3)☐ Information Disclosure Statement(s) (PTO/SB/08)
   Paper No(s)/Mail Date _____.
4)☐ Interview Summary (PTO-413)
   Paper No(s)/Mail Date. _____.
5)☐ Notice of Informal Patent Application
6)☐ Other: _____.

DETAILED ACTION

Continued Examination Under 37 CFR 1.114

1.   A request for continued examination under 37 CFR 1.114, including the fee set

forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this

application is eligible for continued examination under 37 CFR 1.114, and the fee set

forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action

has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on

4/21/2008 has been entered.

Claim Rejections - 35 USC § 101

2.   35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of
> matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the
> conditions and requirements of this title.

Claims 18-22 are rejected under 35 U.S.C. 101 as being directed to non-statutory

subject matter. The language of the claims raises a question as to whether the claims

are directed merely to an environment or machine which would result in a practical

application producing a concrete useful, and tangible result to form the basis of statutory

subject matter under 35 U.S.C. 101.

These claims are rejected because the applicant does not describe the computer

usable data carrier as being tangible medium. Computer readable mediums include

both tangible mediums (storage mediums) and non-tangible mediums (carrier waves

and transmission media). Appropriate correction is required.

To expedite a complete examination of the instant application the claims rejected under U.S.C. 101 (nonstatutory) above are further rejected as set forth below in anticipation of application amending these claims to place them within the four categories of invention.

Claim Rejections - 35 USC § 103

3.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1-8, and 12-22 rejected under 35 U.S.C. 103(a) as being unpatentable over USP, 5,778,387, Wilkerson et al. (Wilkerson hereinafter) in view of USP 6,615,224, Lewis B. Davis (Davis hereinafter).

With respect to claim 1,

WILKERSON teaches method in a data-processing system for recovering data (see Fig. 3, WILKERSON), comprising:

identifying desired data from a command line interface displayable (Fig. 55, Wilkerson) within a display area of a data-processing system (see col. 11, lines 34-41, Wilkerson);

automatically saving said desired data in a memory location of said data-

processing system, in response to identifying said desired data from said command line

interface (see col. 12, lines 16-24, Figs. 3-8 Wilkerson); and

automatically recovering said data from said memory location of said data-

processing system for display within said command line interface, if said desired data is

inadvertently deleted utilizing command line of said command line interface (see col. 19,

lines 50-56, Fig. 32, Claim 1, Wilkerson).

Wilkerson teaches the elements of claim 1 as noted above but does not explicitly

discloses "deleting said desired data utilizing said command line of said command line

interface."

However, Davis discloses "deleting said desired data utilizing said command line

of said command line interface" as a method for deleting files on a UNIX file system, so

that they may subsequently be undeleted, without any possibility of loss or damage. A

file deleted with the "rm" command, or targeted by the "cp" or "mv" commands, is

deleted simply by marking its directory record "deleted", while its inode and data blocks

are not freed. The "ls" command is adjusted so as not to display files whose directory

records are marked as deleted. A indexed system of deleted-file records of such

deleted files is maintained by the UNIX kernel, such that a record for each deleted file

contains a pointer to the file's inode, a pointer to the file's directory inode, and the file's

deletion time. A deleted file may be undeleted simply by calling an "unrm <file>"

command, which uses a kernel system call to undo "deleted" mark in the file's directory

record. This procedure restores the file completely, because its inode and data blocks

were never freed. The "ls" command again will display the file. The indexed system of

deleted-file records, created over time as files are deleted from the UNIX file system, is

a necessary tool used by the kernel to efficiently remove the oldest deleted files from

the system automatically, without excess system overhead, and without cumbersome

system maintenance procedures required from the system administrator (Davis

Abstract).

It would have been obvious to one of ordinary skill in the art at the time the

invention was made to combine the teaching of the cited references because Davis's

teaching would have allowed Wilkerson to provide a method of file protection on UNIX

platforms during file deletion processes, whereby no system performance is sacrificed

and to enhance UNIX operating system performance, because final destruction of the

oldest deleted files is done in large batches.

Claims 12 and 18 have the same subject matter as of claim 1 and are essentially

rejected for the same reasons as discussed above.

As to claim 2,

WILKERSON teaches displaying said data within said command line interface, in

response to automatically recovering said data from said memory location of said data-

processing system (see col. 12, lines 16-24, Figs. 3-8 Wilkerson).

indicating within said command line interface deletion if said desired data in

response to said desired data being inadvertently deleted using said command line

interface (see col. 19, lines 50-56, Fig. 32, Claim 1, Wilkerson).

WILKERSON teaches the elements of claim 2 as noted above but does not

explicitly teaches "displaying an original file of said desired data within said command

line interface, displaying an original file location of said desired data with said command

line interface."

However, Davis teaches displaying an original file of said desired data within said

command line interface and displaying an original file location of said desired data with

said command line interface as FIG. 6 illustrates a particular example of the directory

block 450 with its records for various files and their inode pointers. The particular file

430 is recorded in its directory block 450 at the location 604 in FIG. 6. The inode

pointer 608 in the file record 604 is displayed in the first field 616 of the directory block

(see Abstract and Col 6, lines 32-37, Davis).

It would have been obvious to one of ordinary skill in the art at the time the

invention was made to combine the teaching of the cited references because Davis's

teaching would have allowed Wilkerson to provide a method of file protection on UNIX

platforms during file deletion processes, whereby no system performance is sacrificed

and to enhance UNIX operating system performance, because final destruction of the

oldest deleted files is done in large batches.

As to claim 3,

WILKERSON teaches the step of utilizing said command line interface to interact with an operating system associated with said data-processing system (see col. 12, lines 16-24, Figs. 3-8 Wilkerson).

WILKERSON teaches the elements of claim 3 as noted above but does not explicitly teaches "displaying with the same window of said command line interface said original file, said original file location, said indication of deletion of said desired data, and said recovered data."

However, Davis discloses displaying with the same window of said command line interface said original file, said original file location, said indication of deletion of said desired data, and said recovered data as (see Figure 6 Davis).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of the cited references because Davis's teaching would have allowed Wilkerson to provide a method of file protection on UNIX platforms during file deletion processes, whereby no system performance is sacrificed and to enhance UNIX operating system performance, because final destruction of the oldest deleted files is done in large batches.

As to claim 4-6,

WILKERSON does not explicitly teaches "wherein said operating system comprises a Linux based operating system, Unix based operating system and Windows-based operating system."

However, Davis discloses wherein said operating system comprises a Linux based operating system, Unix based operating system and Windows-based operating system as (see Col 5, Lines 66-67, Col 6, Lines 1-10 and Col 2, Lines 1-10).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of the cited references because Davis's teaching would have allowed Wilkerson to provide a method of file protection on UNIX platforms during file deletion processes, whereby no system performance is sacrificed and to enhance UNIX operating system performance, because final destruction of the oldest deleted files is done in large batches.

As to claim 7,

WILKERSON teaches the steps of: permitting a user to specify a plurality of rules for recycling said data; recycling said data, in response to user input (see col. 24, lines 30-40, Fig. 45 Wilkerson).

As to claim 8,

WILKERSON teaches the step of prompting said user to specify said plurality of rules for recycling said data through a display of a graphical user interface dialog (see col. 19, lines 10-25, Wilkerson).

WILKERSON teaches the elements of claim 8 as noted above but does not explicitly teaches "specifying the minimum size of said data to be recycled and/or specifying special files/empty directories not to be recycled."

However, Davis discloses specifying the minimum size of said data to be recycled and/or specifying special files/empty directories not to be recycled as (see Col 4, Lines 18-23).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of the cited references because Davis's teaching would have allowed Wilkerson to provide a method of file protection on UNIX platforms during file deletion processes, whereby no system performance is sacrificed and to enhance UNIX operating system performance, because final destruction of the oldest deleted files is done in large batches.

Claims 13-17 and 19-22 have the same subject matter as of claims 2-8 and essentially rejected for the same reasons as discussed above.

Response to Arguments

4.   Applicant's arguments filed 4/21/2008 have been fully considered but are moot in view of the new ground(s) of rejection.

In these arguments applicant relies on the amended claims and not the original ones.

Claims must be given the broadest reasonable interpretation during examination and limitations appearing in the specification but not recited in the claim are not read into the claim (See M.P.E.P. 2111 [R-I]).

Regarding claim 1 applicant argues that Wilkerson and Davis do not disclose "deleting said desired data utilizing said command lines of said command line interface" and "automatically recovering said data from said memory location of said data-processing system for display within said command line interface, if said desired data is inadvertently deleted utilizing said a command line of said command line interface."

In response to the preceding arguments examiner respectfully submits that Wilkerson teaches "automatically recovering said data from said memory location of said data-processing system for display within said command line interface, if said desired data is inadvertently deleted utilizing said a command line of said command line interface" as the IC Delete routine, illustrated in FIG. 32 at step 744, has a secondary panel to protect against accidental deletion. After the initial panel is displayed 746, there is a determination if there are panel errors or an exit request from the user 748. If either occurs, the routine exits to step 144 of FIG. 12 (step 750). If neither occur, the process displays another panel 752, requiring the user to confirm the deletion (see col. 19, lines 50-56, Fig. 32, Claim 1, Wilkerson).

Wilkerson further teaches a method for automatically recovering data from a database comprises researching an identified database and an identified time stamp to provide subsystem information needed for the recovery of a database. The method includes automatically creating initial program control language sequences to locate an image copy of database information that is stored in a database recovery repository. The database information is held in a time log format. The method includes recovering database information by using the located image copy stored in the database recovery

repository and automatically formulating a sequence of job control language for

automatically recovering the identified database information stored prior to the time of

the identified time stamp (see Col. 3, Lines 11-24).

These lines teach that the panels on the interface have a recovery routine which

creates program control language sequences automatically for recovering data from

memory location/database recovery repository which has been deleted/lost/corrupted by

accident.

Davis discloses "deleting said desired data utilizing said command lines of said

command line interface" as "deleting said desired data utilizing said command line of

said command line interface" as a method for deleting files on a UNIX file system, so

that they may subsequently be undeleted, without any possibility of loss or damage. A

file deleted with the "rm" command, or targeted by the "cp" or "mv" commands, is

deleted simply by marking its directory record "deleted", while its inode and data blocks

are not freed. The "ls" command is adjusted so as not to display files whose directory

records are marked as deleted. A indexed system of deleted-file records of such

deleted files is maintained by the UNIX kernel, such that a record for each deleted file

contains a pointer to the file's inode, a pointer to the file's directory inode, and the file's

deletion time. A deleted file may be undeleted simply by calling an "unrm <file>"

command, which uses a kernel system call to undo "deleted" mark in the file's directory

record. This procedure restores the file completely, because its inode and data blocks

were never freed. The "ls" command again will display the file. The indexed system of

deleted-file records, created over time as files are deleted from the UNIX file system, is

a necessary tool used by the kernel to efficiently remove the oldest deleted files from the system automatically, without excess system overhead, and without cumbersome system maintenance procedures required from the system administrator (Davis Abstract).

These lines teach working with the command lines such as "rm," "cp," or "mv" commands for the deletion of files/desired data. These deleted files are not being displayed but they are maintained/stored in the Unix kernel. When these files are being undeleted using a recovery command these file are displayed again.

Therefore the combination of the teachings of Davis and Wilkerson provide the invention as a whole because Davis deletes the desired data using a command line and stores the desired data in Unix kernel and when undeleted/recovered it displays it back and Wilkerson teaches automatic recovery by automatically recovering data from memory location/database recovery repository which has been deleted/lost/corrupted by accident.

Applicant further says that instead of removing a filename and pointer from a directory upon deletion in Davis, a deleted flag in a directory block record for the file is set but the record still remain there. In response examiner respectfully submits that claim limitation is broad and does not recite that deletion has to remove a filename and pointer from a directory upon deletion

Contact Information

5.   Any inquiry concerning this communication or earlier communications from the examiner should be directed to Usmaan Saeed whose telephone number is (571)272-4046. The examiner can normally be reached on M-F 8-5.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Hosain Alam can be reached on (571)272-3978. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Usmaan Saeed
Patent Examiner
Art Unit: 2166

Hosain Alam   US
Supervisory Patent Examiner   July 03, 2008

/Hosain T Alam/

Supervisory Patent Examiner, Art Unit 2166